1-1-2015

On using wireless power transfer to increase the max flow of rechargeable wireless sensor networks

Tengjiao He
University of Wollongong, th877@uowmail.edu.au

Kwan-Wu Chin
University of Wollongong, kwanwu@uow.edu.au

Sieteng Soh
Curtin University of Technology, s.soh@curtin.edu.au

On using wireless power transfer to increase the max flow of rechargeable wireless sensor networks

Abstract

A key problem in Rechargeable Wireless Sensor Networks (WSNs) is determining the maximum amount of data that can be collected by a sink over a given time period. This maximum is constrained by link capacity and critically, by the available energy at each node. In this paper, we consider a novel approach to increase the maximum flow rate by exploiting recent advances in Wireless Power Transfer (WPT). Specifically, we deploy a finite number of WPT capable rovers next to bottleneck sensor nodes with the aim to increase the max flow rate of a WSN. We formulate a Mixed Integer Linear Programming (MILP) to determine the routing and the set of sensor nodes that are to be 'upgraded' in order to achieve the maximum flow rate. We also outline a novel heuristic, called Path, to place rovers in large scale WSNs. Our results show it is able to attain on average 85.9% of the optimal flow rate.

Keywords

flow, rechargeable, sensor, networks, transfer, increase, power, wireless, max

Disciplines

Engineering | Science and Technology Studies

Publication Details

T. He, K. Chin & S. Soh, "On using wireless power transfer to increase the max flow of rechargeable wireless sensor networks," in Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2015 IEEE Tenth International Conference on, 2015, pp. 1-6.

On Using Wireless Power Transfer to Increase the Max Flow of Rechargeable Wireless Sensor Networks

Tengjiao He, Kwan-Wu Chin
School of Electrical, Computer and Telecommunications Engineering
University of Wollongong, NSW, Australia
Email: th877@uowmail.edu.au, kwanwu@uow.edu.au

Sieteng Soh
Department of Computing
Curtin University of Technology, WA, Australia
Email: s.soh@curtin.edu.au

***Abstract*—A key problem in Rechargeable Wireless Sensor Networks (WSNs) is determining the maximum amount of data that can be collected by a sink over a given time period. This maximum is constrained by link capacity and critically, by the available energy at each node. In this paper, we consider a novel approach to increase the maximum flow rate by exploiting recent advances in Wireless Power Transfer (WPT). Specifically, we deploy a finite number of WPT capable rovers next to *bottleneck* sensor nodes with the aim to increase the max flow rate of a WSN. We formulate a Mixed Integer Linear Programming (MILP) to determine the routing and the set of sensor nodes that are to be "upgraded" in order to achieve the maximum flow rate. We also outline a novel heuristic, called *Path*, to place rovers in large scale WSNs. Our results show it is able to attain on average 85.9% of the optimal flow rate.**

I. INTRODUCTION

Recently, Wireless Power Transfer (WPT) has had a significant breakthrough. In [1], Kurs et al. reported the powering of a 60W light bulb located two meters away via resonant magnetic coupling with an efficiency of 40%. The authors of [1] also use the same technique to charge multiple devices at the same time. Interestingly, efficiency improves with increasing number of devices. For example, when the charging distance is 200 centimeter, the efficiency of charging two devices is about 68%. This exciting breakthrough means any battery-powered systems can be recharged wirelessly and conveniently.

One such system is rechargeable Wireless Sensor Networks (WSNs). Specifically, we consider a rechargeable WSN with a sink and static sensor nodes that can replenish their battery from energy harvesting technologies such as solar. In addition, these sensor nodes can also be recharged via WPT. Consequently, any "bottleneck" nodes, i.e., those with low energy, can be recharged or "upgraded" in order to achieve a higher flow rate. We note that upgrades will be required if after deployment, sensor nodes have a low energy harvesting rate or have a high energy consumption rate. Consider the example WSN shown in Figure 1, where the aim is to increase the maximum flow rate at the sink. Node *A* and *B* are sources that generate 5 and 3 pkt/s respectively. Nodes *C*, *D* and *E* are able to forward 2 pkt/s each. The capacity of each link is 10 pkt/s. Therefore, the max flow of the network is 2 pkt/s. The reason is that insufficient energy at node *C*, *D* and *E* restricts the max flow at the sink. Now consider the case where a rover with WPT is deployed next to either node *A*, *B*, *C*, *D* or *E*. If the rover is deployed near node *A* or *B*, the max flow at the sink is constrained by the energy available at node *C*, *D* and *E*. If the rover is deployed near node *C* or *D*, the sink only receive 2 pkt/s because node *E* can forward 2 pkt/s. Finally, if the rover is deployed near node *E*, node *C* and *D* can forward 2 pkt/s from node *A* and *B* respectively. Node *E* has sufficient energy to forward packets such that the max flow at the sink is 4 pkt/s. From this example, we see that the key challenge is the finite number of rovers, and selecting the correct subset of sensor nodes in which to park the rovers.

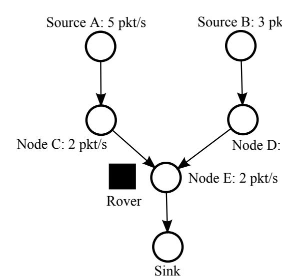

Fig. 1. An example rover deployment to obtain the maximum flow in a rechargeable WSN.

This paper contains the following contributions. We consider the problem of placing multiple rovers with WPT capability to upgrade sensor nodes with the goal of improving the max flow of rechargeable WSNs. We present a Mixed Integer Linear Programming (MILP) for the problem at hand. As the problem is intractable for large scale WSNs, we provide a heuristic called *Path*. Our results show that *Path* is able to attain 85.9% of the optimal flow rate.

Next, Section II reviews previous works on wireless charging and node placement. Section III outlines our network model. In addition, we also describe the problem and present our MILP. Section IV present the details of *Path*. In Section V, we report our results. Finally, Section VI concludes the paper and provides future research directions.

II. Related Works

We first review works that employ mobile chargers (MCs) in WSNs. After that, we consider works that place nodes to improve lifetime and connectivity. We remark that there are also works that aim to maximize flow rate in energy harvesting or rechargeable WSNs; see [2]. However, these works do not employ MCs or rovers. They rely solely on finding a routing that yields the maximum flow rate.

To date, many works have used one MC to prolong the lifetime of a WSN. The main problem is recharging the maximal number of sensor nodes whilst minimizing the traveling distance of a MC. For example, Zhao et al. [3] aim to find the maximum number of recharging points and a tour length that does not exceed a given threshold. The authors of [4] use the future energy consumption rate of nodes when deploying a MC. Shi et al. [5] aim to maximize the ratio of the Wireless Charger Vehicle (WCV) vacation time over a renewable cycle time. This ensures a MC does not waste significant amount of energy on travelling. Similarly, in [6], the authors aim to reduce the distance travelled by MCs. Li et al. [6] aim to find a charging tour that covers the maximum number of nodes. There are also works that consider multiple MCs, especially in large scale WSNs. The main problem is designing a collaborative protocol to prolong network lifetime. For example, in [7] the MCs collaborate to maximize the ratio between the energy used for charging nodes and traveling cost. In a different work, Wang et al. [8] aim to find the minimum number of MCs to achieve perpetual operation. Lastly, the authors of [9] aim to place static chargers in a manner that allows mobile nodes to recharge frequently.

The previous works mainly aim to prolong network lifetime or achieve energy neutral operation. We, however, study methods for placing rovers next to rechargeable nodes with the aim of maximizing flow rate. Note that we assume MCs or rovers are fixed for a given duration T, and we leave the problem of jointly optimizing the distance travelled by MCs and nodes selection with the aim to attain the maximum flow rate as a future work. Apart from that, existing works assume all nodes only have one or two energy sources: battery and/or energy harvesting; e.g., solar. In contrast, this paper considers an additional source: WPT. To the best of our knowledge, no work has addressed the same problem as ours.

Another related research area is node placement. The main aim is to place nodes, e.g., sinks, in order to improve the performance of a WSN. Gianni et al. [10] aim to select the location for relay nodes on a grid to maximize throughput and reduce end-to-end delays. In [11] the authors aim to find the optimal location of cluster headers (CHs) such that the throughput is maximized. The authors of [12] use mobile routers to characterize wireless link quality and also help relay data from sensor nodes to a base station. Deng et al. [13] aim to maximize the collected data of sensor nodes by deploying sinks in an online fashion, whereby during deployment, each sensor node has no information about number, position and data capacity of a sink.

In summary, references [10][11][12][13] consider only one source of energy; i.e., nodes' battery. We consider nodes with WPT capability. References [10][11] and [12] deploy rovers to ensure connectivity and to aid data transfer from sensor nodes to a base station. In [13], the deployed nodes are used to collect data from sensor nodes. However, we use rovers to increase the energy of a subset of sensor nodes.

III. Preliminaries

Before describing our problem, we first introduce the notations used in subsequent sections. We model a rechargeable WSN as a directed graph *G(V, E)*, where each node $i \in V$ is a sensor node, and each link $(i, j) \in E$ represents an edge from node *i* to *j* and has the same link capacity *C*. We define the term Δ as a node's degree. We use $\mathcal{S}$ to represent the sources in a WSN. Let $N_i^- \subset V$ be a set containing the neighbors of node *i* from which it receives data. Conversely, the set $N_i^+ \subset V$ contains the next-hop neighbors in which node *i* transfers data. A special node $s \in V$ is designated as the sink. Each node *i* has a battery with level denoted as e_i. Sensor nodes also have two other energy sources. The first is from their solar panel that provides E_i Joule of energy. Note that the re-charging rate of each node is different [14]. The second energy source is from WPT [15], which provides *B* Joule of energy. Each WPT is placed on a mobile rover. There are γ rovers. We note that the rovers are static after deployment. In addition, they are able to recharge themselves via solar. Let A_i denote the total energy harvested by a node *i* from both environmental and WPT; i.e., $A_i = E_i + B$. We thus have $e_i \leq A_i$. In this paper, the energy consumption rate of each node is only due to communication cost; i.e., the total incoming and outgoing flow rate [16].

As mentioned in Section I, we aim to deploy γ rovers to upgrade the capacity of at most γ sensor nodes to maximize the flow rate from sources to a sink. In other words, extra energy is allocated to γ sensor nodes strategically in order to increase the maximum flow of a WSN. We call the sensor nodes that are recharged by a rover as *bottleneck* nodes.

We will now show that our problem involves solving an NP-hard, *network upgrade* problem, repeatedly. We say a node is upgraded if a rover is parked next to it. In general, using the formalism in [17], when a node is upgraded, then the delay d of its incident links is reduced by $x \times d$, where $x \in [0, 1)$. If both ends of a link are upgraded, then its delay reduces to $x^2 \times d$. In our discussion to follow, we will make use of the following NP-hard, network upgrade problem called *ShortestPath(x, δ)*[17]: given a graph *G(V, E)*, a threshold δ and x, find the minimum number of nodes to upgrade such that the path delay of all node pairs (u, v), where $u, v \in G$, is less than or equal to δ. Assume we are given a max flow rate r bps at the sink with Δ incident links. To achieve this rate, all links $e \in E$, must have a transmission rate that is at least $R_\Delta = \frac{r}{\Delta}$ bps. Each bit then incurs a delay of $1/R_\Delta$ over each hop. Given a network diameter D, we have $\delta = D \times 1/R_\Delta$. Let us set x to the marginal increase in rate, or equivalently, a reduction in delay, if a node is given B Joules of energy. The optimal max flow rate r of a given WSN with γ rovers can then be determined by embedding ShortestPath(x, δ) within a binary search. Specifically, in each iteration, r is adjusted as per the rule of the binary search depending on whether ShortestPath(x, δ) returns more or fewer γ upgrades are required to meet the threshold δ. Consequently, determining the optimal max flow rate requires us to find the optimal number of rovers for a given rate r, an NP-hard problem.

A. Mathematical Model

We now use MILP to model the problem. The first constraint ensures each node must not consume more than its available energy within period *T*. Let $\mathcal{E}$ represent the maximum energy harvested by each node within period *T*. If a rover is parked next to node i, then it receives an extra *B* Joule within period *T* via WPT. Energy consumption is dependent on communication cost. Specifically, a node consumes ρ Joule to receive a bit and τ Joule to transfer a bit. The term $f_{i,j}$ represents the number of bits that node *i* transfers to node *j*. Therefore, the battery constraint is as follows,

$$\rho \sum_{u \in N_i^-} f_{u,i} + \tau \sum_{v \in N_i^+} f_{i,v} \leq R_i B + E_i, \quad \forall i \in V \tag{1}$$

In Equ. 1, the expression $\rho \sum_{u \in N_i^-} f_{u,i} + \tau \sum_{v \in N_i^+}$ corresponds to the energy consumed by node *i* for receiving and transmitting bits. The expression $R_i B + E_i$ represents the available energy at node *i*, including an extra B Joule if a rover is deployed next to it. The decision variable $R_i \in \{0, 1\}$ denotes whether there is a rover next to node *i*.

The next constraint is the standard flow conservation constraint. To ensure this constraint is respected by all nodes, we use the standard process whereby a virtual node is added and is connected to all sources and the sink. The capacity of the edges that connect the virtual node is set to infinity. Therefore, we have

$$\sum_{u \in N_i^-} f_{u,i} = \sum_{v \in N_i^+} f_{i,v}, \quad \forall i \in V \tag{2}$$

Next, Equ. 3 ensures the total flow is within each link's capacity, i.e., *C*.

$$f_{i,u} \leq C, \quad \forall i \in V, \forall u \in N_i^+ \tag{3}$$

The last constraint is to limit the number of deployed rovers to γ. Specifically,

$$\sum_{i \in V} R_i \leq \gamma \tag{4}$$

The objective is to obtain the maximum flow at the sink subject to battery, flow and link constraints. That is,

$$\begin{array}{rl} \text{MAX} & \sum_{i \in N_s^-} f_{i,s} \\ \text{s.t.} & (1), (2), (3), (4) \end{array}$$

It can be shown that the formulated MILP has $2|V| + |E| + |\mathcal{S}| + 3$ constraints, and $|V| + |\mathcal{S}| + |E| + 1$ decision variables. The main difficulty in solving the MILP is the $|V|$ binary integer variables that correspond to candidate placement location for the γ rovers. The search space is of size $\binom{|V|}{\gamma}$. Consequently, for large scale WSNs, increasing γ causes the problem to become intractable quickly. This means the MILP model is solvable only for small instances.

IV. Solution

We now outline a heuristic called *Path* to efficiently determine the location of rovers for large scale WSNs. It has the following key steps. It uses Yen's algorithm [18] to obtain the $|\mathcal{S}|$ shortest routes from the virtual source node to the sink and records the routes in $\mathcal{P}$ in increasing order of their length. Routes are charged according to their length. Specifically, *Path* first recharges the nodes on the shortest route. After placing a rover next to each node on this route, if there are remaining rovers, *Path* recharges the nodes on the next shortest route until all rovers are deployed. In particular, let $\mathcal{P}_a$ be the set of all nodes on route $a \in \mathcal{P}$. We use the term *t* to record the number of remaining rovers that are available for deployment. If *t* is larger than the number of nodes that can be recharged, we deploy rovers next to all nodes on route *a*. The next shortest route is then processed in a similar manner. In contrast, if *t* is smaller than the number of nodes that can be recharged, we deploy all rovers to nodes with the minimal energy. After *Path* deploys the rovers, we increase the set of nodes with a rover accordingly and use a Linear Programming (LP) solver to obtain the max flow.

Referring to Algorithm 1, we now explain the steps of *Path* in detail. *Line 1* obtains $|\mathcal{P}|$ shortest routes from the virtual source to the sink via Yen's algorithm, and sorts them in increasing order of their path length. *Line 2* uses *t* to record the number of remaining rovers. *Lines 3-32* of Algorithm 1 deploy rovers route by route. Specifically, *Line 4* checks the number of remaining rovers. It there is no rover left, *Path* exits; see *Line 30*. As a node may be on several routes, it may be recharged on another route. Thus, *Lines 5-11* exclude the nodes that have already been recharged on other routes. Specifically, *Line 5* uses the term *c* to record the number of nodes on the current route that have a rover placed next to them, and *Lines 6-11* check the current recharging energy of each node on a route and determine the number of nodes that can be recharged, i.e., $|\mathcal{P}_a| - c$. *Lines 12-28* deploy rovers on the current recharging route. There are two cases. First, if there are more rovers than nodes, then we deploy rovers near all nodes that can be recharged; see *Lines 12-20*. Specifically, *Lines 13-19* deploy rovers and record the bottleneck nodes in *R*. *Line 20* updates the number of remaining rovers. In contrast, in the second case, if there are more nodes that need recharging than rovers, we deploy all rovers near nodes with minimal energy; see *Lines 21-28*. Specifically, *Line 22* sorts the nodes in route *a* in ascending order of their energy. *Lines 23-26* deploy the remaining rovers near the *t* nodes with the minimal energy on the route *a*. *Path* has the following property.

Algorithm 1: Path Algorithm

Input: Number of rovers γ. Current energy E_i of each node i
Output: Bottleneck nodes R

1 $\mathcal{P}$=Yen(G(V, E));
2 $t = \gamma$;
3 **for** $a = 1$ **to** $|\mathcal{S}|$ **do**
4 **if** $t > 0$ **then**
5 $c = 0$;
6 **for** $i = 1$ **to** $|\mathcal{P}_a|$ **do**
7 $m = \mathcal{P}_{a_i}$;
8 **if** $E_m \geq B$ **then**
9 $c = c + 1$;
10 **end**
11 **end**
12 **if** $t \geq |\mathcal{P}_a| - c$ **then**
13 **for** $i = 1$ **to** $|\mathcal{P}_a|$ **do**
14 $m = \mathcal{P}_{a_i}$;
15 **if** $E_m < B$ **then**
16 $E_m = E_m + B$;
17 $R_m = 1$;
18 **end**
19 **end**
20 $t = t - |\mathcal{P}_a| + c$;
21 **else**
22 $p = sort(\mathcal{P}_a)$;
23 **for** $j = 1$ **to** t **do**
24 $R_{p_j} = 1$;
25 $E_{p_j} = E_{p_j} + B$;
26 **end**
27 $t = 0$;
28 **end**
29 **else**
30 *Path* exits;
31 **end**
32 **end**

Proposition 1. *The time complexity of* Path *is* $\mathcal{O}(|\mathcal{S}||V|^3 + |\mathcal{S}|\gamma)$.

Proof: The time complexity of Yen's algorithm is $\mathcal{O}(|\mathcal{S}||V|^3)$ [19]. In the worst case, *Lines 3-32* repeat $|\mathcal{S}|$ times to deploy rovers. Next, *Lines 6-11* will check the energy of up to $|V| - 1$ node. Thus, the time complexity of *Lines 6-11* is $\mathcal{O}(|V| - 1)$. *Lines 13-19* at worst repeat $|V| - 1$ times. Given that the shortest route can have at most $|V| - 1$ nodes, the time complexity of *Line 22* is $\mathcal{O}((|V|-1)log(|V|-1))$. Next, *Lines 23-26* repeat for at most γ times. The time complexity of *Lines 23-26* is $\mathcal{O}(\gamma)$. Thus, the time complexity of *Lines 21-28* is $\mathcal{O}((|V| - 1)\log(|V| - 1) + \gamma)$. In summary, the time complexity of *Path* is $\mathcal{O}(|\mathcal{S}||V|^3 + |\mathcal{S}|\gamma)$. ∎

V. Evaluation

The experiments are conducted in Matlab [20] and Matgraph [21]. The parameters used in our experiments are from the following systems. All sensor nodes are MicaZ [16]. We use the theoretical capacity of 250 Kbps, which corresponds to the data rate of the TI CC2420 transceiver [16]. However, in practice, due to protocol overheads, such as channel contention, the actual data rate is likely to be less than 250 Kbps. Hence, the max flow results reported in Section V-A should be interpreted as the theoretical maximum. The TI CC2420 transceiver consumes 209 *nJ/b* and 226 *nJ/b* for receiving and transferring one bit respectively. Therefore, a node consumes 435 *nJ* to forward one bit. In addition, each sensor node is equipped with an Enocean ECS310 solar cell [22]. According to [23], its recharging rate is 150 *mW* in direct sunlight and 1.5 *mW* in cloudy days. Thus, assuming a TI CC2420 transceiver, a node is capable of forwarding up to 350 *kb* per second. The rover is assumed to be a Powermat Charging Spot 2.0 [24], which has an output charging voltage of 6 *V* and its charging efficiency is near 100%. Therefore, a node that is charged by a rover can forward 14000 *kb* per second.

Path is compared against three other approaches: *Random*, *minE* and *LP-round*. The former approach randomly selects nodes to deploy γ rovers. Next, *minE* deploys rovers near the nodes with minimal energy. Specifically, *minE* records the initial energy of each node. Then, it sorts the initial energy of all nodes from the lowest to the highest. Next, *minE* deploys γ rovers next to the first γ nodes with the minimum energy. The last approach, *LP-round*, first removes the integer restriction from the MILP before calling Matlab's LP solver. After that, it sorts the R_i from the highest to the lowest, and deploys rovers to the γ-th highest values in R_i.

A. Results

We study the influence of four parameters: number of nodes $|V|$, number of rovers γ, degree of graph Δ and number of sources $|\mathcal{S}|$. In the following sections, we vary one parameter whilst keeping the others fixed.

We first study varying Δ values; namely 2, 3, 4, 5 and 6. We set the parameters γ, $|\mathcal{S}|$, $|V|$ and $\mathcal{E}$ to 3, 3, 30, 150 *mJ* respectively. Figure 2 shows the max flow of *MILP*, *minE*, *Path* *LP-round* and *Random*. When Δ increases from two to six, the max flow of *MILP* increases from 128 kb/s to 767 kb/s; i.e., an increase of 499.22%. On average, the max flow of *minE*, *Path*, *LP-round* and *Random* is 54.5%, 89.9%, 56.3% and 47.8% that of the *MILP*. This is because of the following reasons. First, if there is only one route from a source to the sink, the flow is limited by the node with the minimum energy. In contrast, as we increase Δ, a source has more neighbors such that the number of routes from the source to the sink increases and thus can forward more data. In addition, increasing Δ also has a positive influence on the max flow of *Path*. As the number of routes increases, the number of nodes on the shortest route from sources to the sink may be equal to or less than the number of rovers. This enables *Path* to deploy all rovers on this route. Therefore, the max flow of the route reaches 250 kb/s, which is the link capacity.

The second experiment investigates the following values of $|V|$: 10, 30, 50, 70 and 90. The value of γ, Δ and $|\mathcal{S}|$ are set

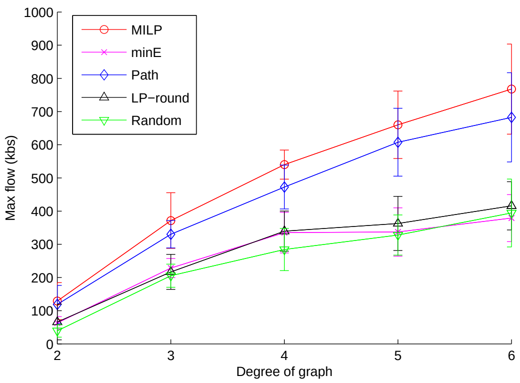

Fig. 2. Max flow with varying Δ.

to three each, and $\mathcal{E}$ to 150 *mJ*. Referring to Figure 3, when $|V|$ increases from 10 to 90, the max flow of *MILP* drops from 665 kb/s to 479 kb/s, or a reduction of 27.92%. On average, the max flow of *minE*, *Path*, *LP-round* and *Random* is 63.4%, 84.5%, 62.1% and 58.1% that of the *MILP*. The reasons for the recorded performance are as follows. As there are only three rovers, this means only three nodes will have a higher energy to forward more data from their neighbors. However, these nodes have little influence on the final max flow in a large network. This is especially significant with increasing $|V|$ as there are more bottleneck nodes.

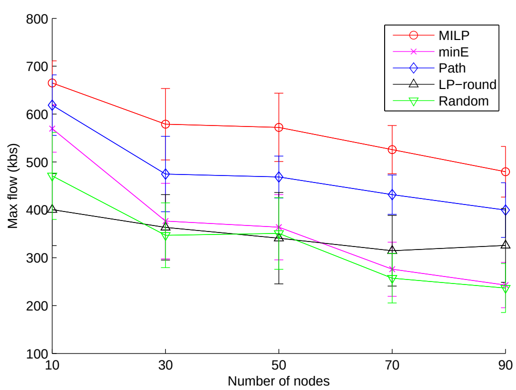

Fig. 3. Max flow with varying $|V|$.

The next experiment investigates the effect of γ where it takes on the following values: 1, 3, 5, 7 and 9. Similar to the last experiment, we set Δ and $|\mathcal{S}|$ to three. Also, $|V|$ and $\mathcal{E}$ have the value of 30 and 150 *mJ* respectively. Referring to Figure 4, when γ increases from one to nine, the max flow of *MILP* increases from 319 kb/s to 745 kb/s. This is an increase of 133.54%. On average, the max flow of *minE*, *Path*, *LP-round* and *Random* is 58.4%, 82.1%, 50.3% and 41.9% that of *MILP*, respectively. As expected, increasing γ leads to more nodes with a higher energy, which helps increasing the max flow. In addition, as the topology remains fixed, increasing γ means more nodes on a path are likely to be recharged, which in turn helps increasing the max flow.

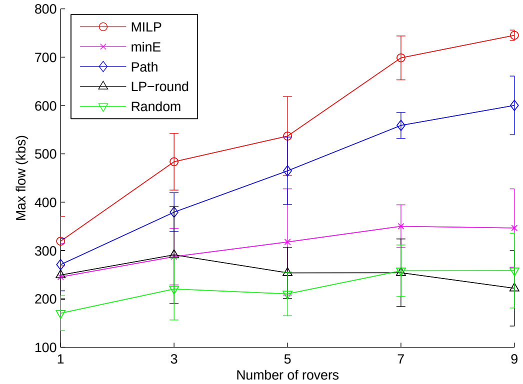

Fig. 4. Max flow with varying γ.

The third experiment studies the effect of $|\mathcal{S}|$ on max flow. Specifically, we set $|\mathcal{S}|$ to the following values: 1, 3, 5, 7 and 9 when parameters Δ, γ, $|V|$ and $\mathcal{E}$ are set to 3, 3, 30 and 150 *mJ*, respectively. We see from Figure 5 that the max flow of *MILP* increases from 253 kb/s to 605 kb/s when $|\mathcal{S}|$ increases from one to nine. For the other algorithms, on average, the max flow of *minE*, *Path*, *LP-round* and *Random* achieves 54.1%, 87%, 51.7% and 49.7% that of the *MILP*, respectively. That is because new sources may be near the sink. Apart from that, sources may be placed on paths with unused energy. For example, those downstream from bottleneck nodes. Lastly, as *Path* uses only $|\mathcal{S}|$ routes, some nodes may not be part of these routes. Consequently, as we add more sources, more of these idle nodes will be used to forward data, and hence increase the flow rate at the sink.

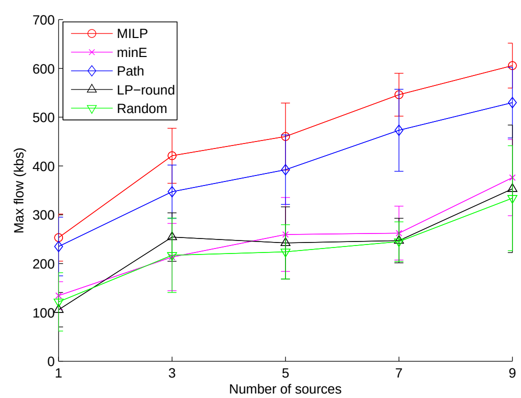

Fig. 5. Max flow with varying $|\mathcal{S}|$.

From our results, there is a maximum of 17.9% gap between *Path* and *MILP*. This is because *Path* always recharges the nodes on the shortest route first. However, it does not consider other routes. As the initial energy of each node is randomly generated, recharging the shortest route first may not generate the max flow for a given topology. *minE* and *Random* do not consider the flow on each node. Thus, the selected nodes may not have a flow. This means with increasing $|V|$ or γ, the probability that *minE* and *Random* deploy rovers near bottleneck nodes decreases. The performance of *LP-round* is close to *minE*. The reason is that *LP-round* may produce rover placements that are very far from the optimal solution.

VI. Conclusion

This paper has investigated the use of WPT-capable rovers to increase the maximum flow rate of a rechargeable WSN. We model the problem succinctly using a MILP, which we also use to compute the optimal solution. We then propose a heuristic called *Path*. Experimental results show that it is capable of achieving on average 85.9% of the max flow rate achieved by the MILP solver. As a future work, we plan to investigate a joint approach that optimizes the trajectory of mobile rovers, their placement, and flow rate. Apart from that, we will consider applying meta-heuristics, e.g., Tabu search, to determine the placement of rovers.

References

[1] A. Kurs, A. Karalis, R. Moffatt, J. D. Joannopoulos, P. Fisher, and M. Soljačić, "Wireless power transfer via strongly coupled magnetic resonances," *Science*, vol. 317, no. 5834, pp. 83–86, 2007.

[2] S. Yang and J. A. Mccann, "Distributed optimal lexicographic max-min rate allocation in solar-powered wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 11, no. 9, p. Articl 9, 2014.

[3] M. Zhao, J. Li, and Y. Yang, "Joint mobile energy replenishment and data gathering in wireless rechargeable sensor networks," in *Procs. of ITC*, (San Francisco, USA), September 2011.

[4] Z. Li, Y. Peng, W. Zhang, and D. Qiao, "J-roc: A joint routing and charging scheme to prolong sensor network lifetime," in *IEEE ICNP*, (Vancouver, BC Canada), October 2011.

[5] Y. Shi, L. Xie, Y. T. Hou, and H. D. Sherali, "On renewable sensor networks with wireless energy transfer," in *IEEE INFOCOM*, (Shanghai, China), April 2011.

[6] K. Li, H. Luan, and C.-C. Shen, "Qi-ferry: Energy-constrained wireless charging in wireless sensor networks," in *IEEE WCNC*, (Paris, France), April 2012.

[7] S. Zhang, J. Wu, and S. Lu, "Collaborative mobile charging for sensor networks," in *IEEE MASS*, (Las Vegas, Nevada, USA), October 2012.

[8] C. Wang, J. Li, F. Ye, and Y. Yang, "Multi-vehicle coordination for wireless energy replenishment in sensor networks," in *IEEE. Intl. Symp on Parallel and Distributed Processing*, (Boston, USA), May 2013.

[9] T.-C. Chiu, Y.-Y. Shih, A.-C. Pang, J.-Y. Jeng, and P.-C. Hsiu, "Mobility-aware charger deployment for wireless rechargeable sensor networks," in *Proceedings the 14th Asia-Pacific Network Operations and Management Symposium*, (Seoul, Korea), September 2012.

[10] E. F. Flushing and G. A. D. Caro, "A flow-based optimization model for throughput-oriented relay node placement in wireless sensor networks.," in *ACM Symposium on applied computing*, (New York,USA), March 2013.

[11] S. Ali, A. Fakoorian, G. Solat, H. Taheri, and A. Eidi, "Maximizing capacity in wireless sensor networks by optimal placement of clusterheads.," in *Canadian Conference on Electrical and Computer Engineering*, (Ontario, Canada), May 2008.

[12] K.-H. Kim, K. Shin, and D. Niculescu, "Mobile autonomous router system for dynamic (re) formation of wireless relay networks.," *IEEE Transactions on Mobile Computing*, vol. 12, no. 9, pp. 1828–1841, 2013.

[13] R. Deng, S. He, and J. Chen, "Near-optimal online algorithm for data collection by multiple sinks in wireless sensor networks," in *IEEE ICC*, (Sydney, NSW), June 2014.

[14] X. Jiang, J. Polastre, and D. Culler, "Perpetual environmentally powered sensor networks," in *IEEE IPSN*, (Los Angeles, USA), April 2005.

[15] L. Xie, Y. Shi, Y. T. Hou, and W. Lou, "Wireless power transfer and applications to sensor networks," *IEEE Wireless Communications Magazine*, vol. 20, no. 4, pp. 140–145, 2013.

[16] J. Suhonen, M. Kohvakka, V. Kaseva, T. D. Hmlinen, and M. Hnnikinen, *Low-power Wireless Sensor Networks: Protocols, Services and Applications*. Springer, 2010.

[17] D. Paik and S. Sahni, "Network upgrading problems," *Networks*, vol. 26, no. 1, pp. 45–58, 1995.

[18] J. Y. Yen, "Finding the k shortest loopless paths in a network," *Management Science*, vol. 17, no. 11, pp. 712–716, 1971.

[19] E. Q. V. Martins and M. M. B. Pascoal, "A new implementation of yen's ranking loopless paths algorithm," *Quarterly Journal of the Belgian, French and Italian Operations Research Societies*, vol. 1, no. 2, pp. 121–133, 2003.

[20] "Matlab." http://www.mathworks.com.au/products/matlab/.

[21] "Matgraph." http://www.ams.jhu.edu/~ers/matgraph/.

[22] "Enocean ecs310 solar cell." https://www.enocean.com/it/enocean_modules/ecs-310/.

[23] S. Sudevalayam and P. Kulkarni, "Energy harvesting sensor nodes: Survey and implications," *IEEE Communications Surveys & Tutorials*, vol. 13, no. 3, pp. 443–461, 2011.

[24] "Powermat." http://www.powermat.com.